

CP/M® VERSION 2.2.04

MANUAL

Volume I



NOTICE

This software is licensed (not sold). It is licensed to sublicensees, including end-users, without either express or implied warranties of any kind on an "as is" basis.

The owner and distributors make no express or implied warranties to sublicensees, including end-users, with regard to this software, including merchantability, fitness for any purpose or non-infringement of patents, copyrights or other proprietary rights of others. Neither of them shall have any liability or responsibility to sublicensees, including end-users, for damages of any kind, including special, indirect or consequential damages, arising out of or resulting from any program, services or materials made available hereunder or the use or modification thereof.

This publication could contain technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of this publication.

Technical consultation is available for any problems you encounter in verifying the proper operation of this product. Sorry, but we are not able to evaluate or assist in the debugging of any programs you may develop. For technical assistance, write:

*Zenith Data Systems Corporation
Software Consultation
Hilltop Road
St. Joseph, Michigan 49085*

or call:

*(616) 982-3884 Application Software/SoftStuff Products
(616) 982-3860 Operating Systems/Languages/Utilities*

Consultation is available from 8:00 AM to 7:30 PM (Eastern Time Zone) on regular business days.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in paragraph (b) (3) (B) of the Rights in Technical Data and Computer Software clause in DAR 7-104.9(a). Contractor/Manufacturer is Zenith Data Systems Corporation of Hilltop Road, St. Joseph, Michigan 49085.

Trademarks and Copyrights

*CP/M is a registered trademark of Digital Research Corporation
Digital Research is a trademark of Digital Research Corporation*

Copyright © 1983 by Zenith Data Systems Corporation.

**HEATH COMPANY
BENTON HARBOR, MICHIGAN 49022**

**ZENITH DATA SYSTEMS CORPORATION
ST. JOSEPH, MICHIGAN 49085**

HOW TO ASSEMBLE YOUR CP/M DOCUMENTATION PACKAGE

In order to make the use of your CP/M products more convenient, you should assemble your CP/M documents according to these instructions.

1. Place the tab labelled "BEGINNING CONCEPTS" in front of the text labelled "Section One: Beginning Concepts" on Page 1-1.
2. Place the tab labelled "SOFTWARE PREPARATION PROCEDURES" in front of the text labelled "Section Two: Software Preparation Procedures" on Page 1-29.
3. Place the tab labelled "REFERENCE GUIDE" in front of the text labelled "Reference Guide" on Page 2-1.
4. Place the tab labelled "APPENDICES" in front of the text labelled "Appendix A: Operating System Error Messages" on Page A-1.
5. Remove the two Digital Research booklets labeled "CP/M 2.0 Interface Guide" and "CP/M 2.0 Alteration Guide" from your existing documentation.

Place the tab labeled "TECHNICAL INFORMATION" on the page labeled "TECHNICAL INFORMATION EXPLANATION" in front of these two booklets.

Preface

CP/M® is an acronym for the Control Program for Microcomputers, the pioneer in the field of microcomputer operating systems. CP/M became instantly popular in the microcomputer field because of its convenience and wide range of practical applications.

This manual is designed to help you make CP/M productive and easy to use in your working environment. Heath and Zenith Data Systems have produced two volumes of CP/M documentation to provide instruction for CP/M users that have different levels of microcomputer experience:

- Volume I, "The CP/M Introductory Guide", provides an overview of CP/M and accompanying software, explanations of concepts for using CP/M with Heath/Zenith hardware, and step-by-step procedures for starting up and preparing your CP/M software.
- Volume II, "The CP/M Reference Guide", provides a comprehensive description of each command included in your CP/M software package.

If you are not familiar with CP/M practices and conventions, you will probably find it useful to read the overview and concept sections of Volume I first. Even individuals with no experience in computers will be able to understand and use CP/M by reading this first volume of the manual.

All users should perform the procedures in Volume I upon receipt of CP/M Distribution Disk media.

When you finish performing the necessary procedures in Volume I, you will be ready to use CP/M with the application program of your choice. But we urge you to make the most of your CP/M software package by exploring "The CP/M Reference Guide" in Volume II. This guide explains all about the many CP/M resident commands and transient commands.

This manual will be more effective if you read it while using your microcomputer. Test each concept on your microcomputer to reinforce your learning.

Table of Contents

| | |
|----------------------|----------|
| Preface | V |
|----------------------|----------|

VOLUME I: THE CP/M INTRODUCTORY GUIDE

| | |
|-----------------------------------------------|------------|
| Section One — Beginning Concepts | 1-1 |
|-----------------------------------------------|------------|

| | |
|------------------------------|------|
| Microcomputer Concepts | 1-3 |
| Disks | 1-3 |
| The Operating System | 1-9 |
| Application Programs | 1-10 |
| CP/M Concepts | 1-12 |
| Files | 1-12 |
| Disk Drives | 1-15 |
| Commands | 1-19 |

| | |
|------------------------------------------------------------|-------------|
| Section Two — Software Preparation Procedures | 1-27 |
|------------------------------------------------------------|-------------|

| | |
|-------------------------------|------|
| Startup Procedures | 1-29 |
| Startup Procedure One | 1-30 |
| Startup Procedure Two | 1-33 |
| Startup Procedure Three | 1-36 |
| Startup Procedure Four | 1-39 |
| Unsuccessful Startup | 1-42 |
| Setup Procedure | 1-43 |
| Introduction | 1-43 |
| Using Setup | 1-45 |
| Troubleshooting | 1-48 |

| | |
|-------------------------------------------|-------|
| Backup Procedures | 1-52 |
| Backup Procedure One | 1-55 |
| Backup Procedure Two | 1-66 |
| Backup Procedure Three | 1-72 |
| Backup Procedure Four | 1-79 |
| Backup Procedure Five | 1-88 |
| Customizing Procedures | 1-94 |
| Customizing Procedure One | 1-96 |
| Customizing Procedure Two | 1-114 |
| Customizing Procedure Three | 1-132 |
| Customizing Procedure Four | 1-145 |
| Customizing Procedure Five | 1-169 |
| Customizing Procedure Six | 1-190 |
| Working Disk Procedures | 1-206 |
| Working Disk Procedure One | 1-208 |
| Working Disk Procedure Two | 1-209 |
| Working Disk Procedure Three | 1-210 |
| Working Disk Procedure Four | 1-211 |
| Working Disk Procedure Five | 1-212 |
| Working Disk Procedure Six | 1-213 |
| System Copying Between Unlike Disks | 1-214 |

VOLUME II: THE CP/M REFERENCE GUIDE
(Provided in a separate binder)

| | |
|-------------------------------------------------------------------------------------------------------------------------------------|-------|
| The CP/M Reference Guide | 2-1 |
| ASM: The Utility that Creates an Intel Hexadecimal File and a Printout File from an Assembly Language Program File | 2-3 |
| ASSIGN: The Utility that Assigns Winchester Disk Partitions to Drive Names | 2-25 |
| BRS: The Utility that Facilitates Backup of Winchester Disk Files to Floppy Disks | 2-33 |
| BSYSGEN: The Utility that Copies the Operating System Between Disks | 2-79 |
| CONFIGUR: The Utility that Customizes the CP/M Operating System for Several Characteristics of the Hardware Environment | 2-87 |
| DDT: The Dynamic Debugging Tool | 2-119 |
| DIR: The Resident Command that Displays Disk File Directories | 2-145 |

| | |
|--------------------------------------------------------------------------------------------------------------------|-------|
| DUMP: The Utility that Displays a File in Hexadecimal Form | 2-149 |
| DUP: The Utility for Copying and/or Verifying Entire Disks | 2-151 |
| ED: The Line Editing Utility that Creates and Edits Text Files | 2-163 |
| ERA: The Resident Command for Erasing Files | 2-185 |
| FORMAT: The Utility that Prepares the Disk Surface | 2-189 |
| LIST: The Utility that Prints File Contents on Paper | 2-203 |
| LOAD: The Utility that Loads a "HEX" File for Execution | 2-211 |
| MAKEBIOS: The Utility that Customizes the Operating System for Disk Drives | 2-215 |
| MOVCPMxx: The Utilities that Customize a CP/M Operating System Kernel to Fit Memory Size and Disk Type | 2-221 |
| PIP: The Utility that Copies Data Between Files, Disks, and/or Devices | 2-227 |
| PREL: The Utility that Creates a Relocatable File from Two Hexadecimal Output Files | 2-247 |
| REN: The Resident Command that Renames Files | 2-251 |
| SAVE: The Resident Command that Copies Data from Memory to a Disk File | 2-253 |
| SETLP: The Utility that Temporarily Sets Printer Characteristics | 2-257 |
| STAT: The Utility that Reports Disk Statistics and Assigns Types of Status | 2-263 |
| SUBMIT: The Utility that Triggers Automatic Execution of CP/M Commands | 2-277 |
| SYSGEN: The Utility that Puts the Operating System on a Disk | 2-283 |
| TYPE: The Resident Command that Displays File Contents on the Console | 2-289 |
| USER: The Resident Command that Controls User Access to File Areas | 2-291 |
| XSUB: The Utility that Batches Commands Within Utility Programs for Automatic Processing | 2-295 |

Appendices

Appendix A: Operating System Error Messages A-1

Appendix B: Recommended Hardware for
Heath/Zenith CP/M B-1

Appendix C: Bootstrap C-1

Appendix D: Basic Input/Output System
(BIOS) Organization D-1

Index

Section One

Beginning Concepts

This part of the manual explains concepts that are important when using the CP/M Operating System within your Heath/Zenith microcomputer environment.

The concepts explained here are grouped under the headings:

- “Microcomputer Concepts” Provided for individuals who have never before used microcomputers, and for individuals with limited microcomputer experience who wish to review.
- “CP/M Concepts” Provided for individuals who are new or unfamiliar with the CP/M Operating System.

Individuals who are proficient at using both microcomputers and CP/M can skip these parts of the manual, and proceed directly to the “Software Preparation Procedures”.

EXAMPLES OF USER/COMPUTER DIALOG

This text contains examples of user interaction with a microcomputer. In these examples, displays presented on the microcomputer terminal will be represented by the following typestyle:

THIS TYPESTYLE represents terminal displays

0123456789#*\$?: = . A> ()

User input (the characters that you type through the terminal) will be represented by boldface type, as shown:

BOLDFACE TYPE represents the things you type

0123456789#*\$?: = [.](0

Text that instructs you to press a key labelled with more than one character (such as the RETURN key or the CTRL key) will specify the label of this key in dark (boldface), slanted (italic), capitalized characters, as shown:

Press ***RETURN***

Hold down ***CTRL***

In many instances, the exact text of a display will vary by a few characters. This manual often substitutes a few letters in place of exact characters where variations are likely to occur. For instance, this manual will illustrate a program's serial number as "Serial number sss-sssss", while your terminal might actually display it as "Serial number 357-81469".

In cases where the exact characters you type will vary, this manual presents a description of the necessary characters within curved braces, {}.

Hence, this manual might explain that an entry should be made in the following form: **B:=A:{filename.ext} RETURN**, when you actually type the characters **B:=A:CONFIGUR.COM RETURN**.

Hardware device model numbers beginning with the "H/Z-" prefix are references to either a Heath device, a Zenith device, or both. For example "H/Z-89" in this manual refers to hardware devices that are labelled either "H-89" or "Z-89".

MICROCOMPUTER CONCEPTS

Your microcomputer is a sophisticated piece of equipment that reflects the latest technical advances in the computer field. But this machine is practically useless without the programs of instruction that tell it what to do.

These programs are stored on disks and used by your microcomputer when they are required to perform a task. This section explains how the programs are stored on disks, shows you how to handle your disks, and defines two important types of program: the operating system and the application program.

Disks

Stored information, or data, is arranged in concentric rings on the surface of a disk. These rings are called "tracks". Each track is divided into areas called "sectors". Each sector contains data measured in units called "bytes". A byte of data could be one letter typed at the terminal keyboard or one instruction in a program. But since a byte is such a small unit, you will more often see data measured in "kilobytes". A kilobyte (abbreviated as "k") is equal to 1024 bytes.

Data is transferred to disks in the form of magnetic impulses generated by an electromagnet called the "read-write head". As the name implies, the read-write head can read data from the disk or write data on the disk — similar to the way in which the head in a tape recorder transfers magnetic impulses to and from recording tape.

But unlike a tape recorder, a disk drive unit can transfer data at any location on the disk surface almost instantly, because the drive is usually spinning the disk at a high rate of speed. Whenever the read-write head is instructed to read or write data at a particular location on the disk, it positions itself along the appropriate track and skims across the surface of the disk as the appropriate sector spins by. Each disk has a directory that tells the read-write head which track and sector it should access to transfer the necessary information in the proper sequence.

There are two different kinds of tracks on every disk: system tracks and file tracks. System tracks are reserved for the disk's directory and for part of your operating system, and they are usually the two outermost tracks on the disk. File tracks are reserved for files, and they are the inner tracks on the disk. (Upcoming text will explain the concepts behind operating systems and files.)

NOTE: Before data can be written on a disk, the surface of the disk must be prepared. Disk preparation is performed by a program called "FORMAT", which is stored on your Distribution Disk (or Distribution Disk I). FORMAT prepares the disk surface by dividing it into tracks and sectors. The procedures for constructing backups and customizing the operating system will show you how to prepare disks using FORMAT.

Heath/Zenith offers disk drive units that will run the CP/M system with both floppy disks and Winchester Disks. Upcoming text deals with the unique aspects of each of these disk types.

FLOPPY DISKS

A floppy disk is a circular sheet of mylar plastic that is coated with a magnetic oxide and contained within a square plastic cover.

Floppy disks, and the data stored on them, are fragile. Therefore, you should adhere to the following precautions to ensure that disks and stored data are not damaged.

Floppy Disk Handling Precautions

- When holding the disk, touch only the protective square disk cover. Do not touch the brownish disk surface that shows through the read-write access slots in the disk cover.
- Keep disk in the protective paper envelope whenever it is not within a disk drive.
- Do not allow dust, ashes, liquid, or any other foreign material to contact the disk surface.
- Keep disk away from electric motors, appliances, telephones, etc., as these devices contain magnets that could alter the magnetic impressions on the disk.

- Never put a disk into a drive unit before turning on hardware equipment; and never leave a disk in a drive unit while the power is being turned off. Sudden fluctuations in the power supply to your hardware environment could cause the read-write head to “crash” against the disk surface, and destroy stored information.
- Do not expose disks to temperatures above 125 degrees Fahrenheit (52 Centigrade), or temperatures below 40 degrees Fahrenheit (10 Centigrade).
- Never press a ball-point pen or a pencil directly against the cover of a disk. Instead, mark disk labels before adhering them to the disk cover, or mark them using a felt-tip pen while they are on the disk cover.
- Do not allow the disk or its cover to be bent, creased, or torn.

Write Protecting and Write Enabling Floppy Disks

You can mechanically prevent or allow the writing or erasing of information to or from your disks by covering or uncovering the notch in the disk cover with specially-provided tabs. The way that you use these tabs depends on the size of the disk.

5.25-inch Disks

With 5.25-inch disks, the notch is covered to **prevent** you from writing to or erasing from the disk. Therefore by putting the tab on a 5.25-inch disk you are “write protecting” the disk. A 5.25-inch disk with a notch that is not covered can be written to or erased from. Therefore a 5.25-inch disk with an uncovered notch is “write enabled”. Figure 1-1 illustrates this distinction.

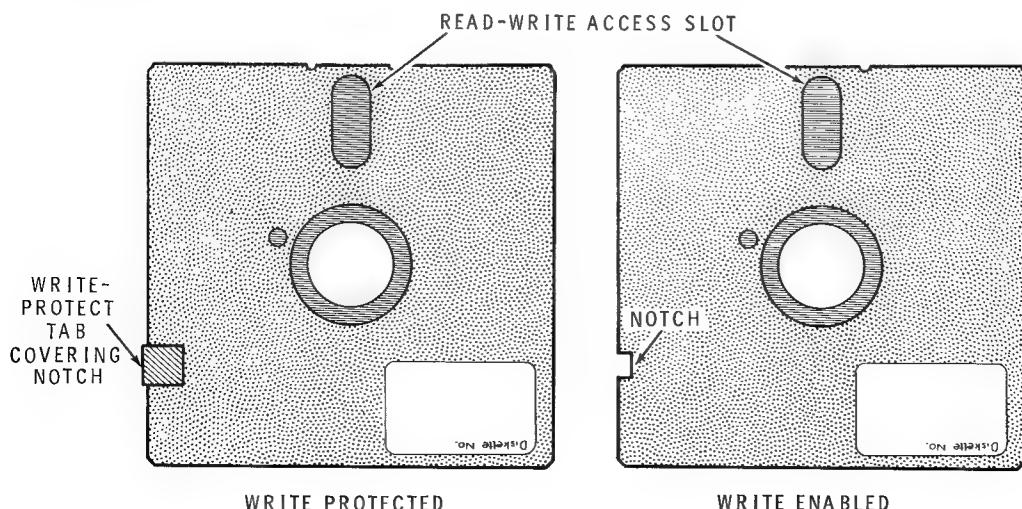


Figure 1-1
5.25-inch Floppy Disks

8-inch Disks

With 8-inch disks, the notch is covered to allow you to write to or erase from the disk. Therefore by putting the tab on an 8-inch disk you are "write enabling" the disk. An 8-inch disk with a notch that is not covered can not be written to or erased from. Therefore an 8-inch disk with an uncovered notch is "write protected". Figure 1-2 illustrates this distinction.

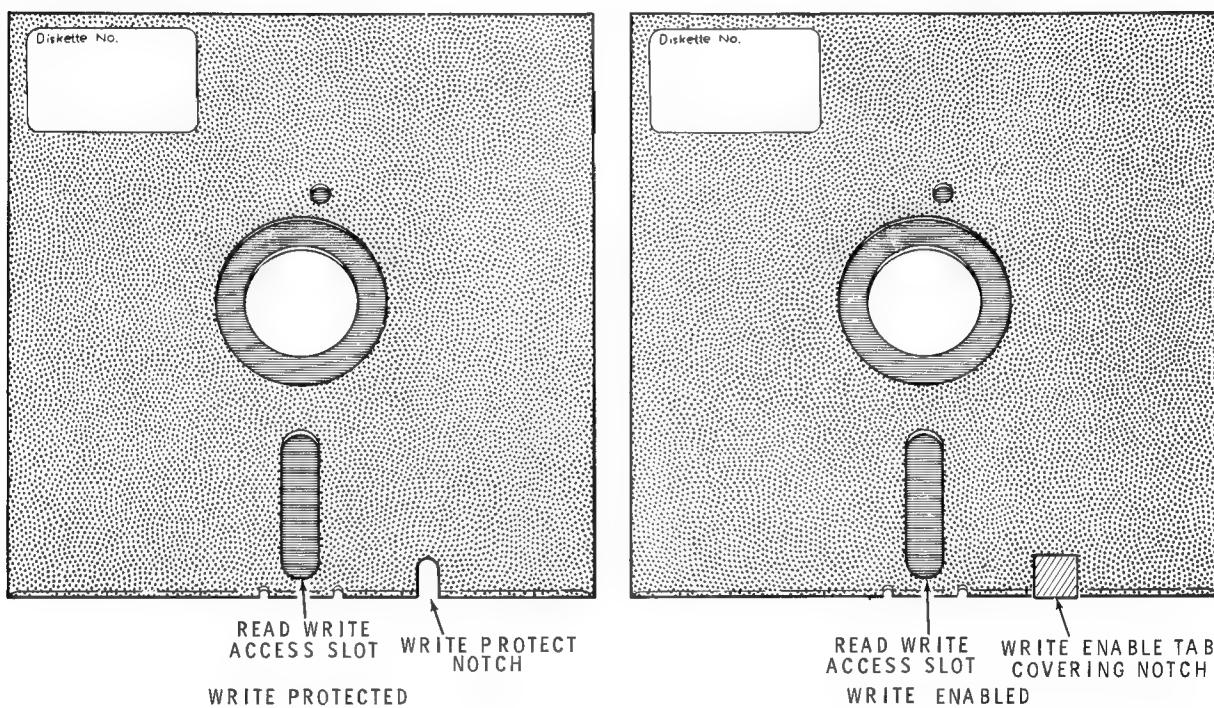


Figure 1-2
8-inch Floppy Disks

Eight-inch disks can also be mechanically write protected if you turn "on" a "write-protect" switch corresponding to each drive that contains a disk that you wish to protect. These switches are located on the front panel of the H/Z47 or H/Z67 disk drive models. You must always perform a warm boot after changing the setting of a write-protect switch. You can perform a warm boot by holding down the **CTRL** key and pressing the **C** key.

NOTE: Whether a disk is write protected or write enabled, you can read or copy data from it.

Inserting Floppy Disks

To insure that data stored on your floppy disks can be safely and efficiently accessed, the disks must be inserted into drives carefully and correctly.

When inserting a floppy disk into a disk drive, the read-write access slot in the square plastic cover should be pointing towards the back of the drive. A label is usually affixed to one side of the plastic disk cover, and this label should face upward or to the left as the disk is inserted into the drive. When the disk is fully inserted in the drive slot, close the drive door or fastening latch.

NOTE: For all floppy disk drive units, the power must be turned on **before** the disk is inserted. Floppy disks should **never** be within a drive when the power is off.

Figure 1-3 illustrates the proper technique for inserting floppy disks into some of the many drives that Heath/Zenith offers.

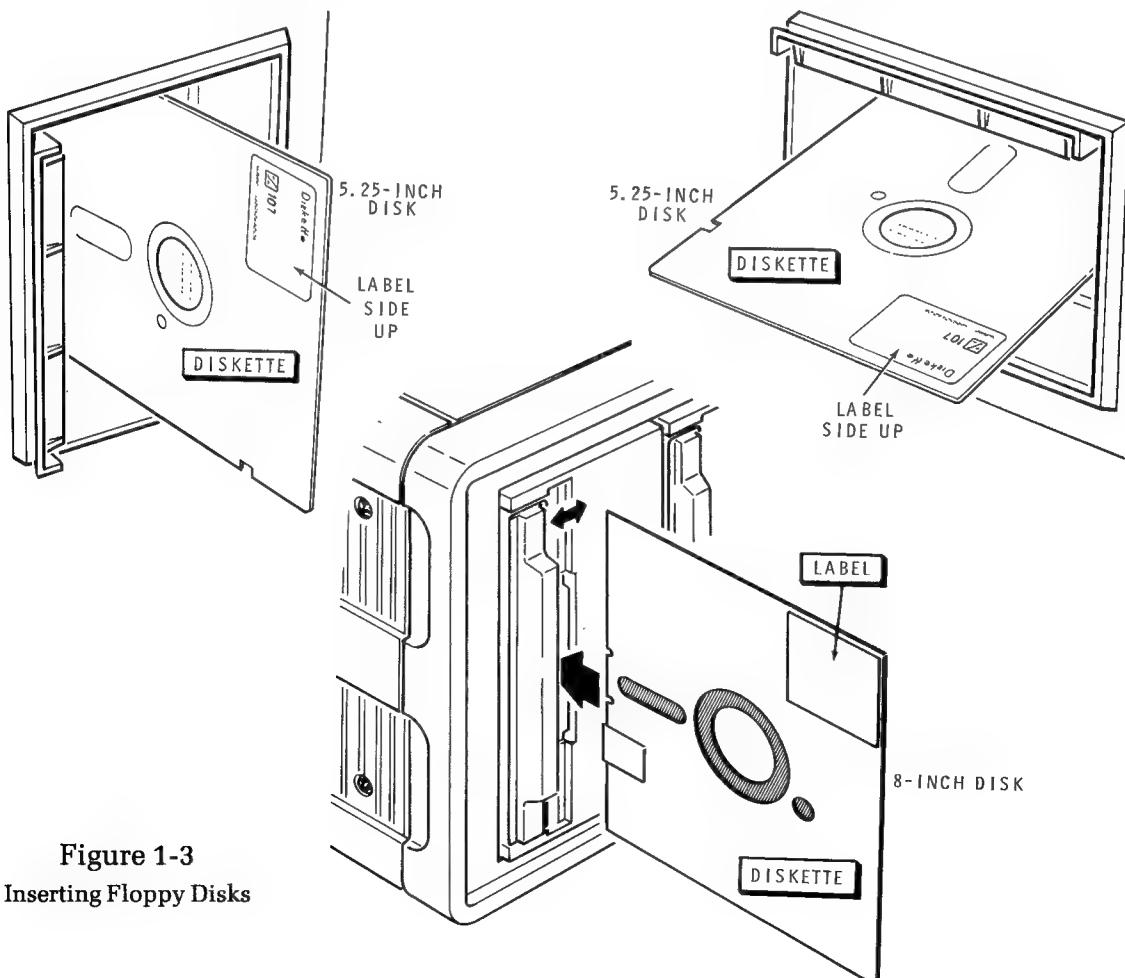


Figure 1-3
Inserting Floppy Disks

WINCHESTER DISKS

A Winchester Disk is a round metal plate with a coating of magnetic oxide on its surface. This disk is enclosed in the left-hand chamber of the H/Z-67 drive model.

Because the Winchester Disk has such a great capacity for stored data, it comes with a special program that allows you to establish separate data areas, called "partitions", on the disk. Winchester Disk partitions act as though they are separate disks. You can establish up to 62 distinct partitions on one Winchester Disk, and access them by using the AS-SIGN program stored on your CP/M Distribution Disk.

Since Winchester Disk partitions behave so much like disks, we will refer to them as "disks" throughout much of this manual.

NOTE: The H/Z-67 drive unit also contains a drive that accommodates 8-inch floppy disks. Information about such disks can be found in the text section dealing with "Floppy Disks".

Winchester Disk Precautions

Because the Winchester Disk stores so much information within such small surface areas, you should adhere to the following precautions to ensure that the disk and stored data are not damaged.

- Keep electric motors, appliances, telephones, etc. away from the H/Z-67 drive unit, as these devices contain magnets that could alter the magnetic impressions on the disk.
- Do not expose the H/Z-67 drive unit to temperatures above 125 degrees Fahrenheit (52 Centigrade), or temperatures below 40 degrees Fahrenheit (10 Centigrade).

Write Protecting and Write Enabling the Winchester Disk

The Winchester Disk can be mechanically set to prevent the writing or erasing of information to or from it, by turning "on" the left-hand "write-protect" switch (switch "0") on the front of the H/Z-67 drive unit.

NOTE: You must always perform a warm boot after changing the setting of this switch. You can perform a warm boot by holding down the **CTRL** key and pressing the **C** key.

The Operating System

An operating system is a computer program that controls both the components of your hardware environment and subordinate programs ("application programs") that perform specific tasks.

It provides a vital link between your keyboard and your application programs, and between your application programs and your peripheral hardware. Thus it is essential that you use an operating system whenever you use an application program in your microcomputer environment.

The way that you use the CP/M Operating System is to copy an image of it (a "system image") from a disk and put it inside the memory of your microcomputer. This activity (called "bootstrap") is explained in the text entitled "Startup Procedures".

CP/M can perform several useful functions when it is alone in your computer's memory, but it also has "hollow" areas in which you can also store application programs as you need them.

OPERATING SYSTEM COMPONENTS

The Heath/Zenith CP/M operating system is divided into two software components: the "system kernel" and the file "BIOS.SYS". A disk that contains both of these components can be said to contain the CP/M operating system. Such a disk can also be said to be "bootable", which means that it contains an image of the operating system that can be inserted into the computer and used.

The System Kernel

The "system kernel" is a set of instruction programs that reside on the system tracks of a disk. These disk tracks are reserved for the system kernel and the disk's directory. Special data transfer methods are necessary to move the system kernel from one disk to another. The system kernel helps to translate your keyboard entries into input that the computer can use.

The BIOS.SYS File

The "BIOS.SYS" file is a set of instruction programs that reside on any of the file tracks of the disk. BIOS.SYS can be manipulated by the same methods used to manipulate other files that are stored on

the disk. "BIOS" stands for Basic Input/Output System, which is the part of the operating system that enables CP/M to work with Heath/Zenith hardware.

Operating System Requirements

To use the operating system, you must transfer a copy of it from a disk to the computer by performing an activity called "bootstrap" or "bootup" or "cold booting". Both the system kernel and the BIOS.SYS file **must** be on a disk for that disk to be usable for bootup (the activity that puts CP/M in the computer) or for any other CP/M-related activities.

Furthermore most users must change, or "customize", both the system kernel and the BIOS.SYS file so that they will be compatible with the hardware devices that they must control. The text entitled "Customizing Procedures" will show you how to make the appropriate changes to a copy of the CP/M Operating System.

Application Programs

An application program is a set of instructions that tells your microcomputer how to perform a specific function.

Application programs are stored on the file tracks of a disks in units of data called files, (as explained in the "Files" text section). An application program might consist of several files that automatically access each other under certain circumstances. Whenever an application program file is needed to perform a specific task, an image of this file is copied from the disk and inserted into computer memory, where the CP/M Operating System has reserved "hollow" memory space for it.

After the application program files have served their purpose, CP/M moves them aside and either reserves its hollow memory areas for new application programs or executes some of the programs within the operating system itself.

EXAMPLES OF APPLICATION PROGRAMS

Your Distribution Disk contains several application programs which are often referred to as "transient commands" or "utilities". These programs are stored on the Distribution Disk, and are identified by names

that end with the file extension "COM". This extension indicates that these files are valid COMmands, that can execute under CP/M.

The following list shows just some of the other kinds of application programs that can perform specific tasks when controlled by the CP/M Operating System.

BUSINESS:

Electronic Spread Sheet
Inventory Management
General Ledger
Accounts Receivable
Accounts Payable
Sales Invoicing
Payroll
Client Posting & Accounting
Property Management
Word Processing
Mail List Management
Electronic Dictionary
Data Base Management
Automatic Letter Generator

PROGRAMMING LANGUAGES:

BASIC-80 Interpreter
BASIC-80 Compiler
C-BASIC
FORTRAN-80
COBOL-80

SYSTEM UTILITIES

Macro Assembler
Printer Spooler
Symbolic Debugger

APPLICATION PROGRAM REQUIREMENTS

It is important to remember that an application program cannot produce the results you desire without the presence of an operating system. Therefore, you must always load the CP/M operating system into your microcomputer before you can use any application program. (The text entitled "Startup Procedures" will show you how to put CP/M in your computer.)

Furthermore, an application program cannot use all of the devices in your hardware environment unless the CP/M Operating System that controls it has been customized to work with your hardware. (The Customizing Procedures will show you how to customize CP/M.)

CP/M CONCEPTS

This text section explains some of the basic properties of the CP/M Operating System and some of the conventions you must follow when using the system. After reading Microcomputer Concepts and this section, you should be ready to perform the Software Preparation Procedures.

Specifically, this section shows you how CP/M stores data in units called files, how CP/M accesses these files through disk drives, and how you should issue commands to the CP/M Operating System.

Files

The CP/M Operating System enables you to create, analyze, and manipulate data by storing this data in units called "files". These files are stored on the surface of a disk and given a name that conforms to CP/M file naming conventions.

When you issue a command that refers to a particular file by name, CP/M goes to the appropriate disk, makes a copy of the file, and puts the copy into one of the hollow memory areas inside the computer.

CP/M FILE NAMING CONVENTIONS

A file name consists of two parts: the primary name and the extension. The primary name has between one and eight characters, and is essential in all file names. The extension can have between one and three characters, or it can be omitted entirely. The primary name and the extension are separated by a period (.), in the following form:

{primary name}.{extension}

The characters used in the primary name and extension can be any character on the console keyboard except the following special characters:

<>,,: =?*[]

The following example file names all conform to these conventions:

FORMAT.COM memo.doc BIOS.SYS 4/27/81.TXT
 JOB3.HEX FILE#1 33%-RATE.DAT WSMSGS.OVR

Although the extension is optional, you'll probably find it useful to give your files extensions that somehow describe the type or purpose of the file you are naming. The following list shows several extensions that are applied to files used for a specific purpose:

| EXTENSION | FILE PURPOSE |
|-----------|------------------------------------------------|
| COM | Command binary file (executable under CP/M) |
| ASM | Assembler source file |
| HEX | Intel HEX object file |
| PRN | Print file of program listing |
| BAS | BASIC source file |
| INT | Intermediate BASIC file (for BASIC-E, C-BASIC) |
| BAK | Backup file from prior edit |
| LIB | Library source file |
| SUB | A list of commands to be executed with SUBMIT |
| \$\$\$ | Temporary work file |
| FOR | FORTRAN source file |
| DAT | ASCII data file |
| DOC | ASCII document file |

REFERENCING SEVERAL FILES AT ONCE (WILDCARD FILE NAMES)

Many of the commands that you will issue refer to files by name. But when you want to issue the same command for several files with similar names, it is often more convenient to enter a "wildcard file name" with the command.

A "wildcard file name" represents several file names — much like a "joker" in a deck of cards can stand for any card in the deck. (Wildcard file names are sometimes called "ambiguous file names".) A wildcard file name contains either asterisks (*), or question marks (?), or both.

An asterisk (*) is entered in place of the entire primary name and/or extension of a file, as shown in the following example:

16MAY82.*

In this example, the asterisk replaces the extension so that this wildcard refers to files on the disk with the primary name “16MAY82” and **any** extension.

In the following example, the wildcard asterisk takes the place of the primary name:

*.COM

Therefore, all files with the extension “COM” and **any** primary name would be referenced.

The wildcard “*.*” stands for **any** file on the disk. (Actually, there are exceptional circumstances under which all of the files on a disk cannot be referenced at one time, but these circumstances will be explained in later text.)

Question marks (?) can be used in a wildcard file name to take the place of single characters at fixed file name positions. For instance, the wildcard file name

JOB?.HEX

would refer to any file that has the extension “HEX” a primary name with the characters “J”, “O”, “B”, and one or fewer additional characters. Hence the files “JOB0.HEX”, “JOB1.HEX”, “JOBY.HEX”, and “JOB.HEX” are just a few of the files that could be referenced by such a wildcard (if these files existed on the disk).

You can use any number of question marks in a wildcard (up to 11) and the actual file names referenced will be those with the same characters as those that are explicitly stated in the wildcard and any characters in the place of the question mark characters.

Thus the wildcard “?????????.COM” will reference any file on the disk with the “COM” extension — just like the “*.COM” wildcard.

Disk Drives

A disk drive is a device that transfers data to and from disk storage media — whether that media is on a floppy disk or on a Winchester Disk partition. (Since floppy disks and Winchester Disk partitions are so similar in function, this text will refer to both as “disks” in most cases.)

DRIVE NAMES

To allow you to refer to disks and files within your disk drives, the CP/M Operating System recognizes each drive in your hardware environment by a distinct “drive name”. A drive name consists of a letter of the alphabet in the range “A” through “F”, and a colon (:).

Possible drive names are “A:”, “B:”, “C:”, “D:”, “E:”, and “F:”.

DEFAULT DRIVE

The drive from which you perform bootstrap is always referred to as “default drive A:”, as indicated by the “A>” system prompt that your console displays when the CP/M image is inside the computer. The default drive is the drive to which the system will refer, unless you specifically tell the system to refer to a different drive.

You can execute an application program that is stored as a file in the default drive by typing the **primary** file name of the application program file (the part without the “COM” extension) in response to the system prompt, as shown:

A>{primary name}

Where “A>” is the default drive prompt (or system prompt); and

where **{primary name}** is the primary name of the file that you want to execute. For this command to be valid, the file must reside on a disk in default drive A:, and have a “COM” extension.

If the Application Program file represented by **{primary name}** does not reside on default drive A:, then the system will display a message repeating your unfound entry with a question mark, as shown:

A>{primary name}
{primary name}?

This kind of error message will also occur if you use improper syntax in your entry or misspell your entry.

Changing the Default Drive

You can change the default drive by typing the name of another drive and a carriage return at the “A>” prompt, as shown:

A>B: RETURN

Such an entry will produce a new system prompt, indicating that drive B: is now the default drive, as shown:

A>B: RETURN
B>

NOTE: Any drive that is changed to the default drive in this fashion must be a valid drive within your hardware environment, and it must contain a floppy disk prepared by the FORMAT program or a Winchester Disk partition prepared by both the FORMAT and ASSIGN programs.

You can experiment with changing default drives by entering drive names (with a carriage return) one at a time at the system prompt. When you switch default drives in this fashion, the CP/M system will assume that any Application Program you wish to execute should be found on a disk in the new default drive. This experiment is a good way to determine the names of your drives if you’re not sure.

Accessing a Nondefault Drive (Logging In)

A “nondefault drive” is a disk drive whose name is not displayed in the system prompt. For instance, if the “A>” system prompt is displayed (meaning A: is the default drive), then your nondefault drives are any valid drives with names “B:” through “F:”.

When you want to execute an application program that resides in a disk file in a nondefault drive, type the name of the appropriate nondefault drive immediately before typing the program’s primary name and a carriage return:

A>B:{primary name} RETURN

Where "A>" is the default drive prompt (or system prompt); where B: is the name of the desired nondefault drive; and where {primary name} is the primary name of the file that you want to execute. The file must reside on a disk in nondefault drive B:, and have a "COM" extension.

The CP/M system would respond to such an entry by "logging in" disk B: to get the application program file indicated by {primary name}, inserting an image of this program into computer memory, and executing the program.

NOTE: If you try to reference a drive that does not exist in your hardware environment or a drive that does not contain a disk (by logging in, or switching default status), then the system will look for the referenced disk drive indefinitely. Meanwhile, the system will accept no other commands unless you reset the computer and perform a cold boot.

SWITCHING FLOPPY DISKS BETWEEN DRIVES

When you reference a floppy disk drive (by entering a command at a default drive, by changing default drives or by logging in a nondefault drive with a command), the CP/M system remembers some of the characteristics of the disks in the referenced drives. Switching disks between drives can cause problems unless you tell the system to forget the old disks.

You can make the system forget about old disk characteristics by performing a "warm boot". You can perform a warm boot by holding down the CTRL key and pressing the C key. This entry is often abbreviated as **CTRL-C**. This entry is usually entered in response to the "A>" system prompt. It tells the operating system to forget what it knew about the disks that used to be in the drives and to redisplay the system prompt.

We suggest that you perform a warm boot whenever you remove a disk from a drive and replace it with another disk (unless the system prompts you to insert a different disk).

NOTE: If you perform a warm boot during the execution of some application programs, execution of this program might end, whereupon control would return to CP/M.

FOR USERS WITH ONE OR TWO 5.25-INCH DRIVES

A hardware environment with one or two 5.25-inch drive slots appears to have just that many drives. But under the CP/M Operating System, such a disk drive environment enables you to perform most of the activities you could perform with three drives.

CP/M actually believes that you have three disk drives. The three drives that CP/M believes you have are called "logical" drives. The drives that you can actually see in front of you are called "physical" drives.

When you enter a command that requires more disks than you have physical drives, CP/M displays prompts that instruct you to insert the required disks at the appropriate time.

The prompts will appear in the form:

PUT DISK B IN DRIVE A: AND PRESS RETURN

When such a prompt appears, you should remove the disk that is already in the drive, insert the disk that the prompt indicates, and enter a carriage return. Execution of the program will resume until data from a different disk is needed. Then a similar prompt will appear, requesting that you insert a different disk.

When switching disks between a drive in this manner, you do not have to perform a warm boot after a disk switch. However, you must keep track of which disk is which. If you insert the wrong disk, you might have to start the activity over.

Procedures for constructing backup disks and customizing the operating system contain instructions for putting this "logical/physical drive" concept to work.

NOTE: CP/M's logical/physical drive feature enables you to use many utilities that transfer, change, or create data amongst two or three disks in a single activity. However, the DUP utility cannot be used at all unless your number of physical drives matches the number of disks required for the DUP activity.

Commands

In general, a command is a program that can help you to create, change, analyze, or move data. Commands are entered in response to a “system prompt”.

A system prompt consists of the letter for the default drive and the greater-than (>) character. When you start up CP/M, the system prompt is displayed on your console, as shown:

A>

The system prompt tells you that CP/M is ready to receive a command in the form of a “command line”.

COMMAND LINES

A command line is the form of response you make to the system prompt to bring up, or “invoke”, a command. A command line usually consists of three components: the “function”, the “argument”, and the “carriage return”. The function is entered first, and it indicates the activity that will be performed. The argument is entered one space after the function. The argument indicates what data (files, systems, disks, drives, etc.) the function’s activity should be performed upon. After entering the function and argument, you must enter a carriage return to tell CP/M that the entire command line is ready for execution.

You will enter command lines in the following form:

A>{function} {argument} **RETURN**

Where “A>” is the system prompt;

where {function} is mandatory for all commands;

where {argument} is optional for some commands; and

where **RETURN** is mandatory for all commands.

Always separate the command line function and the command line argument with one space. Furthermore, any command entered at a prompt with the “>” character must end with a carriage return. However, commands themselves often display prompts as well. When such

a prompt ends with a colon (:), a carriage return is not required in your response.

NOTE: In this text, the entry of a carriage return will often be illustrated in the form **RETURN**.

There are two kinds of commands that can be executed in a CP/M operating environment: Resident Commands and Transient Commands.

RESIDENT COMMANDS

Resident Commands reside within the CP/M operating system. Therefore, CP/M doesn't have to refer to a disk to know how to execute these commands — although the commands themselves might affect data that is on a disk.

The CP/M Operating System contains the following resident commands:

| | |
|------|------------------------------------------------------------------------------------|
| DIR | Displays the names of files that reside on a disk. |
| ERA | Erases specified files from a disk. |
| REN | Renames a specified file on a disk. |
| SAVE | Saves the contents of computer memory space by creating a file on a disk. |
| TYPE | Displays the contents of a file on the terminal. |
| USER | Enables you to divide the space on a disk into separate areas for different users. |

This list shows only the command line function of the resident commands. See Volume II: "The CP/M Reference Guide" for a comprehensive explanation of the arguments used when these commands are entered.

TRANSIENT COMMANDS

Transient commands are application programs that are supplied with the CP/M Operating System on your Distribution Disks. These application programs help you to manipulate the operating system and to perform several other useful microcomputer activities.

These commands (also known as "utilities") are stored on the disk as files with the "COM" extension. When you issue a command that makes reference to one of these files, CP/M takes an image of this file from the appropriate disk, inserts this image into one of CP/M's hollow areas inside computer memory, and begins execution of the transient command.

The following list shows the files containing all of the transient commands offered on Heath/Zenith CP/M distribution disk media.

| | |
|--------------|--------------|
| ASM.COM | MAKEBIOS.COM |
| ASSIGN.COM | MOVCPM17.COM |
| BRS.COM | MOVCPM37.COM |
| BSYSGEN.COM | MOVCPM47.COM |
| CONFIGUR.COM | MOVCPM67.COM |
| DDT.COM | PIP.COM |
| DUMP.COM | PREL.COM |
| DUP.COM | SETLP.COM |
| ED.COM | STAT.COM |
| FORMAT.COM | SUBMIT.COM |
| LIST.COM | SYSGEN.COM |
| LOAD.COM | XSUB.COM |

To use a transient command in a command line, you type the primary file name of the file that contains this command. This primary name is the command line function.

Due to the complexity of the activities that these commands perform, some require special files for their command line arguments. The following list shows some of the argument files required by some transient commands:

DUMP.ASM BIOS.ASM MAKEBIOS.SUB

The procedures for making backup disks, customizing the operating system, and making working disks contain step-by-step instructions for entering transient command lines.

Volume II: "The CP/M Reference Guide" supplies further details on all aspects of these commands.

COMMAND LINE ENTRY

The CP/M Operating System is very precise in the way it accepts command lines. You must spell all components of a command line correctly and include the names of nondefault disks drives whenever a referenced file is not on the default disk. If you don't, CP/M will not be able to execute your command and will respond by redisplaying the invalid command line with a question mark (?).

However, CP/M does allow some flexibility in the way you may respond to the system prompt. CP/M performs special functions when you type any of the special entries that are explained below.

NOTE: In this text, "CTRL" followed by a hyphen and a letter indicates that you should hold down the key marked "CTRL" (control key) and then press the key of the specified character.

The following list explains the single keys and combinations of keys that you can press to edit a command line before submitting it to CP/M for execution.

| | |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DELETE | Removes the previous character typed from the command line. Depending on how your Operating System is adjusted, the removed characters might be echoed (repeated in reverse) on the video console display, or erased from the display. |
| DEL | Removes the previous character typed from the command line. Depending on how your Operating System is adjusted, the removed characters might be echoed (repeated in reverse) on the printed console display. |
| BACK SPACE | Removes the previous typed character. Also removes any "DELETED" characters that were echoed in the line. |
| CTRL-H | Same as "BACK SPACE". |
| CTRL-X | Removes all characters typed in the command line, as if you used "BACK SPACE" all the way to the beginning of the line. |
| CTRL-U | Effectively removes all characters typed in the command line, and allows you to try again on the line beneath the old line. It leaves the display of the old command line on |

the console, and displays the “#” character at the end of this old line to label it as a nullified entry.

CTRL-R Redisplays the edited version of a command line below the “scratch pad” version of the line without any of the “DELETED” characters that might have been echoed in the line. Also displays the “#” character at the end of the “scratch pad” version.

The following list explains the single keys and combinations of keys that you can press to end a command line and submit it to CP/M for execution.

RETURN Ends the command line, sends the command to the system for execution, and displays nothing on console. After execution of the command, CP/M redisplays a system prompt. In this manual, you will be instructed to press this key in the form **RETURN**.

CTRL-J Same as “RETURN”.

CTRL-M Same as “RETURN”.

The following list explains the single keys and combinations of keys that you can press to change the way in which CP/M executes your command line:

CTRL-S Interrupts the display of data to the console when pressed once. Allows CP/M to resume data display when pressed a second time. This entry is useful when data scrolls by on a console too quickly for you to read it. Part III: Reference Guide explains the commands during which it is safe and useful to make this entry.

CTRL-P Causes CP/M to send everything it displays on the console device to the list device (LST:) at the same time. (The list device is usually a printer or a modem.) Making this entry a second time will stop the display to the list device. This entry is useful when you want to record the displays that appear during the execution of a command on paper, or send them through the telephone.

The routing of console displays to the list device will continue during and after the execution of any resident command or transient commands STAT, DUMP, and CONFIGUR. This data routing will remain in effect during (not after) the execution of any other transient command except SUBMIT, XSUB, and MAKEBIOS. However, this entry will not cause data routing to the list device during the execution of most other application programs.

If you type a CTRL-P entry while your list device is disconnected, turned off, in a local mode, or off line, then your system might try to route data through the inactive list device indefinitely, and refuse to accept commands until you reset and perform a cold boot.

The following entry enables you to enter an unusually long command line:

CTRL-E Enables you to see the entire display of a command line that is longer than your screen or platen is wide. When you type this entry, the remaining portion of your command line will be displayed at the left-hand end of the screen or platen, below the beginning portion of the command line.

This entry will not send your command line to the system for execution (as a "RETURN" entry would). It is not essential that you enter CTRL-E when typing a command line that exceeds console display range, because CP/M will process your command line even if you cannot see the whole line.

Even if you type CTRL-E entries, no command line can exceed 127 valid characters in length. If you type a 128th character in a command line, CP/M will automatically interpret it as a carriage return and try to execute the first 127 characters as a command.

The following entries enable you to enter comments that CP/M will ignore:

; (The semicolon.) Enables you to enter comments not intended for execution without receiving error feedback from CP/M. To cause CP/M to ignore a comment, you must make the ";" the first character entered at the system prompt.

Comments can consist of any characters you wish, typed after the ";" entry, and followed by a "RETURN" or "CTRL-U" or "CTRL-X".

: (The colon) Same as ";" (semicolon).

The following entries enable you to rapidly skip several spaces in a command line or comment:

| | |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TAB | As with a regular typewriter, this key enables you to advance several spaces without pressing the space bar several times. It skips to the eighth column of the console display range or to some column numbered by a multiple of eight. Hence if you enter a TAB at the beginning of a command line, you will skip six columns (because the system prompt takes up two columns). If you immediately enter another TAB, you will skip eight columns, and so forth. |
| CTRL-I | Same as "TAB". |

Section Two

Software Preparation Procedures

This part of the manual provides five kinds of procedures that will help you prepare your CP/M software so that it works efficiently with your hardware:

- “Startup Procedures” help you to prepare your microcomputer hardware for use, and to load the CP/M Operating System into your microcomputer.
- “Setup Procedure” helps users of most Heath/Zenith hardware environments to copy CP/M distribution software and to make system disks. With proper hardware, Setup can be used instead of Backup Procedures and Customizing Procedures and Working Disk Procedures.
- “Backup Procedures” help you to copy your CP/M distribution software to backup disks, to protect your software investment.
- “Customizing Procedures” help you to adjust the CP/M Operating System so that it controls your hardware environment.
- “Working Disk Procedures” help you to combine useful application programs on a customized bootable disk.

Each procedure begins with a key or table to help you determine which section of the procedure you should use, based on the kind of hardware you have.

EXAMPLES OF USER/COMPUTER DIALOG

This text contains examples of user interaction with a microcomputer. In these examples, displays presented on the microcomputer terminal will be represented by the following typestyle:

THIS TYPESTYLE represents terminal displays

0123456789#*\$?: = . A>()

User input (the characters that you type through the terminal) will be represented by boldface type, as shown:

BOLDFACE TYPE represents the things you type

0123456789#*\$?: = [.]0

Text that instructs you to press a key labelled with more than one character (such as the RETURN key or the CTRL key) will specify the label of this key in dark (boldface), slanted (italic), capitalized characters, as with **RETURN** and **CTRL**.

In many instances, the exact text of a display will vary by a few characters. This manual often substitutes a few letters in place of exact characters where variations are likely to occur. For instance, this manual will illustrate a program's serial number as "Serial number sss-sssss", while your terminal might actually display it as "Serial number 357-81469".

In cases where the exact characters you type will vary, this manual presents a description of the necessary characters within curved braces, { }. Hence, this manual might explain that an entry should be made in the following form: **B:=A:{filename.ext} RETURN**, when you actually type the characters **B:=A:CONFIGUR.COM RETURN**.

Hardware device model numbers beginning with the "H/Z-" prefix are references to either a Heath device, a Zenith device, or both. For example "H/Z-89" in this manual refers to hardware devices that are labelled either "H-89" or "Z-89".

If you have trouble performing a procedure or if you obtain an error message, then refer to Volume II: "The CP/M Reference Guide". The reference guide contains comprehensive explanations of each utility. If you encounter an error message that is not explained in the reference guide, then consult Appendix A: "Operating System Error Messages".

STARTUP PROCEDURES

This section will explain the sequence of steps necessary for starting up a session of CP/M use. This sequence includes the preparation of your hardware devices, the insertion of a bootable disk into the appropriate drive, and the movement of a copy of the CP/M operating system from a disk into your microcomputer's memory.

The most significant step in this sequence is the movement of CP/M from a disk into the microcomputer. This step is known as "bootstrap" or "booting up" or "cold boot". You will perform this step at least once, each time you use CP/M in your microcomputer. Once inside your microcomputer, CP/M can control an application program or perform one of the many tasks within its own repertoire.

The startup procedure you should use is determined by the type of microcomputer you have.

If you have an H/Z-88, H/Z-89, or H/Z-90 microcomputer, then you should use "Startup Procedure One".

If you have an H-8 microcomputer without special options, then you should use "Startup Procedure Two".

If you have an H-8 microcomputer with the PAM-8 Panel Monitor Program, use "Startup Procedure Three".

If you have an H-8 microcomputer with the HA8-8 Extended Configuration Board, use "Startup Procedure Four".

These procedures show you how to boot up using only the first drive of your primary drive group. These procedures also assume that your distribution software is recorded on disks that can be used in the first drive of your primary drive group.

If your distribution software is not recorded on disks that can be used in the first drive of your primary drive group, then you will need to enter a different bootup command than the one advised in your startup procedure or change a switch setting within your computer. Refer to Appendix C: "Bootstrap" for instructions on different bootup commands or switch settings.

NOTE: The CP/M Operating System recorded on your distribution media is preset to work with a terminal that has a baud rate of 9600. The H/Z-88, H/Z-89, and H/Z-90 microcomputers contain terminals with a 9600 baud rate. However, some printing terminals that can be used with the H-8 microcomputer have baud rates other than 9600, which will not allow you to perform a startup procedure until the baud rate of your CP/M system has been changed to match that of your terminal.

Therefore, if you are using a printing terminal with a baud rate other than 9600 (such as a Decwriter or Diablo KSR 1640), you will first need to customize your CP/M system using a different terminal that does have a baud rate of 9600.

Startup Procedure One

H/Z-88, H/Z-89, or H/Z-90 Microcomputer

This procedure consists of five steps to help you start up the CP/M Operating System in your H/Z-88, H/Z-89, or H/Z-90 microcomputer.

1. Connect and power up **all** of your hardware devices.

Refer to the manuals for each device to be sure that it is properly connected and turned on. In general, all of the "on-off" switches should be in the "on" position. When turned on, your microcomputer should display the following message in the upper left-hand corner of the display screen:

H:

2. If this display does not appear after you turn the computer on (allowing time for the screen to warm up), then hold down the **SHIFT** key and press the **RESET** keys. This entry resets your microcomputer.
3. Insert your CP/M Distribution Disk (or Distribution Disk I or the Setup Utility Disk or a different bootable disk) into your primary boot drive. You can determine which drive is your primary boot drive by the following guidelines:
 - If you have a 5.25-inch drive mounted beside your display screen, insert the disk into this drive.